Optimized Diagnosis of Central Nervous System (CNS) Cancer using Gene Expression Microarray & Machine Learning (ML) Methods

Deepak Painuli[*1], Suyash Bhardwaj[2], Utku Kose[3]

[1,2]Gurukula Kangri Vishwavidyalaya, Haridwar, INDIA
[3]Suleyman Demirel University, Isparta, TURKEY
Email Id:- deepak.painuli@gmail.com [*1], suyash.bhardwaj@gkv.ac.in[2], utkukose@sdu.edu.tr[3]

*Corresponding Author: - deepak.painuli@gmail.com

Abstract
Central nervous system (CNS) cancer is among the top 11 causes of cancer-related fatalities worldwide. Classification and early diagnosis of various tumor forms are crucial in CNS cancer analysis to protect patients from mortality. Conventional diagnostic methods, could be subject to high misdiagnosis rate due to inter & intra-observability variations observed in human interventions during diagnosis process. Higher efficiency & lower error rate of machine learning (ML) methods on complex & high dimensional data problems makes ML methods suitable choice for gene expression based diagnosis of CNS cancer. By analyzing the gene expression data, this study's primary goal is to demonstrate the CNS cancer diagnosis using various ML models using efficient feature selection (FS) methods named recursive feature elimination (RFE) & maximum relevance - minimum redundancy (mRMR) and model optimization by grid search method. This study investigates 12 ML models i.e., Logistic Regression (LR), SVM (Linear/RBF), K-nearest Neighbor (KNN), Naive Bayes (NB), Decision-Tree (DT), Random Forest (RF), Extra Tree (ET), Gradient Boost (GbBoost), Extreme Gradient Boost (XgBoost), Adaptive Boost (AdaBoost) and Multi-layer Perceptron (MLP), for CNS cancer diagnosis task to obtain best ML model to accurately classify CNS cancer subjects using CNS cancer gene expression dataset. Experimental and comparative study of previously held research's results demonstrated that LR-based model along outperformed all other applied models with classification accuracy of 99.6%, precision of 0.99, recall of 0.98, F1-score of 0.99 and AUC-Score of 1.0 on RFE-100 feature subset, which observed to be highest among various ML models employed on similar gene expression dataset in recent past.

Keywords: Medical Diagnosis, Central Nervous System Cancer, Machine Learning, Gene Expression, Feature selection, GridSearchCV.

1. Introduction

Undoubtedly, cancer is one of the most dreaded illnesses in the world. The Centers for Disease Control and Prevention (CDC) estimates 2022, cancer accounted for around 609,360 deaths in the United States, making it the second largest cause of death [1] Brain and other nervous system malignancies are among the top 11 causes of cancer-related fatalities, according to the American Cancer Society. [2] Brain and other central nervous system tumors make up roughly 3% of all cancers worldwide but are more common in males than women. Estimates show that 308,102 cases of Central Nervous System (CNS) cancer were discovered worldwide in 2020, resulting in 251,329 fatalities. [3] About 18,600 Americans died from brain and other nervous system cancers in 2021, and there were about 24,530 new instances of the disease nationwide. [4] In India itself, the incidence of CNS tumors is from 5 to 10 per 100,000 people, with an upward trend, and they account for 2% of all malignancies. [5] The phrase "CNS Cancer" refers to a variety of cancers or tumors that develop in the brain or spinal cord and are frequently fatal because of their invasive nature and propensity to reject common surgical techniques and treatments. [6] Although central nervous system cancer has a major social and economic impact on the patients, their families, and the community, Additionally, the intrinsic disability of people with brain and other central nervous system tumors places a tremendous load on healthcare systems. [6-7]

CNS cancer represent a substantial cause of death in the entire world. Classification and early diagnosis of various tumor forms are crucial in CNS cancer analysis to protect patients from mortality.[8] Due to the human involvement or use of rule-based methodologies, traditional CNS diagnosis may be time-consuming and reasonably error-prone. [9] Artificial intelligence (AI) techniques, such as machine learning (ML) techniques, may offer a solution because of their effectiveness and lower error rate than human beings. In the case of high-dimensional gene expression microarray data, it may also be impossible to apply conventional methods of diagnosis. Thanks to advancements in science and technology, it is now possible to use a method named microchip to analyze the expression of hundreds of genes. [10-12] The quantity of information provided by microarray technology on the expression levels of thousands of genes has been used for both diagnostic and prognostic reasons for a variety of disorders. [13] By nature, Gene expression microarray used to be high dimensional data having thousands of features, therefore, high performance classification techniques are crucial for the analysis of massive amounts of data, including microarray gene expression data. [14-15] This high dimensionality brings complexity to considered ML methods during diagnosis process of CNS cancer as well as it adds overfitting issue to ML model. Therefore, before developing diagnosis model, feature dimensions need to be reduced using efficient feature selection and reduction method to ensure optimized diagnosis of CNS cancer without overfitting. [16-18]

The goal of this study is to develop an ML model for the early detection of CNS cancer using gene expression data by first selecting highly corelated features using the Maximum Relevance - Minimum Redundancy (mRMR) feature selection (FS) method [19], adopting the Recursive Feature Elimination (RFE) method [20] to search for top contributor features, and then fine tuning the ML model using Grid Search method. [21]

The further organization of this paper is as follows: the analysis of prior research on CNS cancer diagnosis using ML techniques is supported by Section 2 of the paper. In Section 3, the proposed approach for CNS cancer detection is illustrated. Section 4 contains a discussion and analysis of the results obtained. The conclusion of the paper is provided in Section 5.

2. Literature Review

The classification & diagnosis of CNS cancer based on gene microarray data using ML approaches has recently been the subject of numerous investigations by various scholars. The following are a few of the significant studies and a quick description of each.

Gunavathi. C. et.al. [22] proposed K-nearest Neighbor (KNN) & Support Vector Machine (SVM)-based cancer classification model based on gene expression data of various cancer types. Authors adopted T-Test, F-Test & SNR values-based feature selection method for dimensionality reduction task. Highest accuracy of 81.25 % have been noticed during this study against CNS cancer classification task.

Salem, H. et.al. [23] proposed novel gene expression-based methodology to classify different cancer disease. Information Gain (IG) and Standard Genetic Algorithm (SGA) are combined in the proposed methodology. IG is used initially for feature selection, followed by Genetic Algorithm (GA) for feature reduction and Genetic Programming (GP) for the categorization of different cancer kinds. Proposed methodology scored improved accuracy of 86.67% in CNS cancer classification task.

Arslan. M.T. et. al. [04] investigated multiple ML models i.e., SVM, Multilayer Perceptron (MLP) & Decision Tree (DT) for CNS cancer identification using high dimensional microarray data of CNS cancer gene expression. This study utilized Correlation-based FS (CFS) method for dimensionality reduction and investigation results identified MLP as best classifier with an outstanding accuracy of 97.6%.

Danaee, P. et.al. [24] presented deep learning (DL)-based method for cancer detection using microarray-based gene expression data. Authors analyzed high dimensional gene expression data using Stacked Denoising Autoencoder (SDAE) to thoroughly extract functional features. ML classifiers named Artificial Neural Network (ANN) & SVM were investigated during this study for CNS cancer identification task and improved accuracy of 96.26% was achieved by SVM method.

Adiwijaya. Et.al.[25] presented principal component analysis (PCA)-based CNS cancer diagnosis methodology for high-dimensional gene expression microarray data. Levenberg-Marquardt Backpropagation (LMBP) & SVM algorithm were utilized as classifier and highest accuracy of 93.3% was received by SVM-based ML model during CNS classification task.

Koul. N. et.al. [26] presented a method for gene selection from six publicly available microarray cancer gene-expression datasets using simulated annealing (SA) and various ML models i.e., MLP, Random Forest (RF), SVM & Adaptive Boosting (AdaBoost). Experiment results demonstrated supremacy of MLP-based ML model over rest of utilized ML model withstanding improved accuracy of 96%.

Al-Obeidat. et. al. [27] investigated SVM, KNN & RF-based ML methods for CNS tumor classification using gene expression data. Authors utilized IG and GA methods for efficient feature selection throughout this study. Experiment results identified RF-based ML model as best classifier with an outstanding accuracy of 97.3%.

Kabir MF. et.al. [28] presented Neural Network (NN) & SVM-based methodologies for cancer detection using microarray-based data. This study utilized PCA algorithm for feature selection and dimensionality reduction purpose. Result shows dominance of SVM-based model over NN-based model for CNS tumor classification task with an improved accuracy of 97.8%.

As can be seen from the review above, various ML techniques have been applied in recent research efforts on gene expression data-based classification of CNS cancer. However, it should be noted that most of the recent studies on microarray-based CNS cancer detection hardly investigated either two- or three-ML classifiers. Also, some research used various feature selection (FS) approaches, whereas others did not. Similar to this, most studies don't suggest looking for the best hyperparameters to increase the testing accuracy of a given ML-Model. Taking the aforementioned into consideration, this study examined all 12 of the ML-models on the CNS cancer gene expression dataset and also uses sophisticated FS methods like the mRMR & RFE and Grid Search method to find the optimal hyperparameter of the top three ML-Models i.e., LR, Support Vector Classifier (SVC (Linear)) & SVC (RBF), for improved testing accuracy. Several performance criteria, including accuracy, recall, precision, F1-score, and AUC-score, were considered to assess efficiency of the suggested ML model. To determine the model's effectiveness, these results were also contrasted with those from a number of other ML models that were applied in the previously reviewed studies.

3. Methodology

This study's methodology was composed of several stages i.e., dataset procurement, data preprocessing, data sampling, feature selection, model's pre-evaluation, dimensionality reduction, model selection, model optimization and model's post-evaluation stage as presented in figure-1.

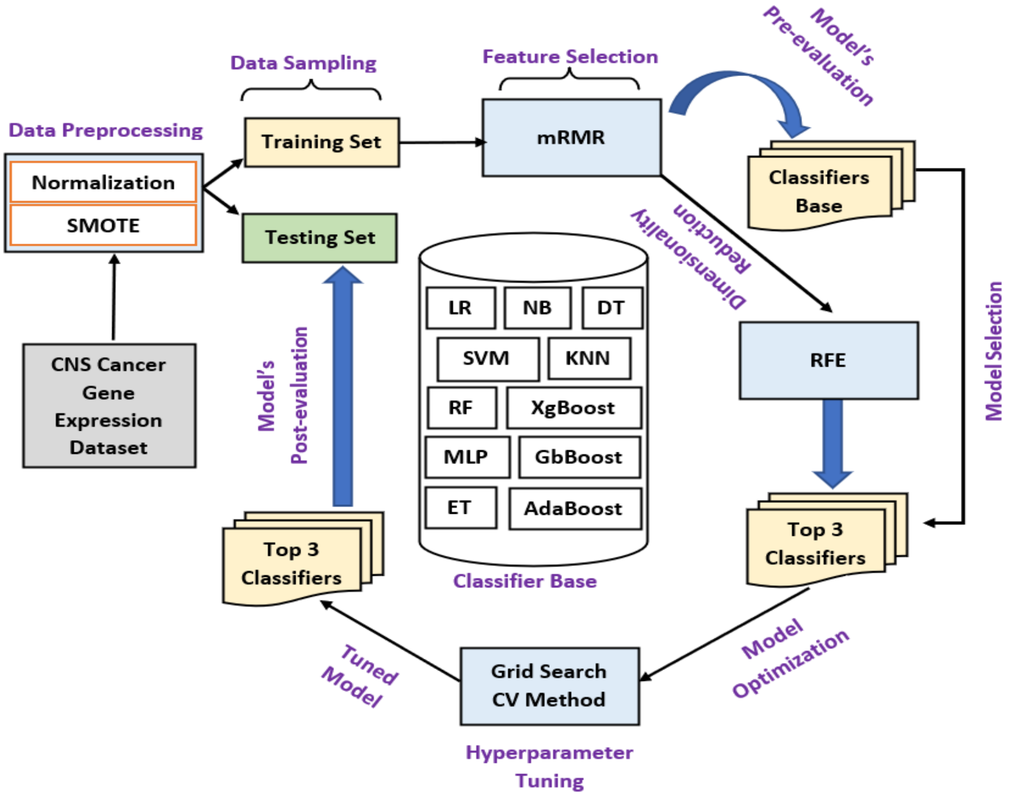


Fig.1. Process flow of proposed method

3.1 CNS Cancer Dataset

CNS cancer data in the form of microarray data were employed in this study to classify CNS cancer. Dataset was procured from Kent-Ridge Biomedical Data Repository Dataset consist of 60 samples of two classes (Class-0=39, Class-1=21), one is “Survivors / Control” class ("Class-1") and “failures / non-Control” class ("Class-0"). Dataset includes total of 7129 features against 60 samples making it high dimensional dataset to work upon, along with that dataset also seems to be imbalance dataset due to observed class imbalance ratio of 1.85. [29-30]

3.2 Data Pre-processing

Two data preprocessing procedures—data normalization and Synthetic Minority Oversampling Technique (SMOTE) were used in this study. When an ML model is used, the former one is often a data preparation technique needed to constrain the feature value scale to a specific range, whilst the latter one deals with class imbalance control to reduce the complexity of the ML model by reducing skewness or biases. [31-32]

3.2.1 Data Normalization

When features are around the same size and/or close to being regularly distributed, ML models perform well and converge more quickly. In order to scale feature values to unit variance and standard deviation in this study without losing the feature's information, we used the Standard Scaler approach, resulting in traits that are normally distributed. [33-34] Computation of standard scaler may be described as follows-

$$Z = \frac{(\chi - \mu)}{\sigma}$$

where,

μ = Training sample’s mean.
σ = Training sample’s Standard deviation.
χ = Training sample’s value.
Z = Standard score / z-scores of training sample

3.2.2 Class Imbalance Control via SMOTE

As observed in section 3.1, CNS cancer microarray dataset seems to be imbalance dataset due to observed class imbalance ratio of 1.85. Imbalanced dataset mostly leads to overfitting as well as biased prediction issues, hence class imbalance issue needs to be resolved before proceeding to future phase of model development. This study utilized Synthetic Minority Oversampling Technique (SMOTE) to oversample minority class’s samples count (Class-1=21) till majority class’s samples count (Class-0=39). SMOTE is an algorithm that adds artificial data points to the actual data points to accomplish data augmentation. [35-36] SMOTE can be viewed as an improved form of oversampling or as a particular data augmentation procedure. With SMOTE, we avoid producing duplicate data points and instead produce synthetic data points that are marginally different from the original data points.

Working principles of SMOTE oversampling techniques are as follows: -

- A random sample is chosen from the minority class.
- Find the k nearest neighbors for the observations in this sample.
- The vector between the current data point and one of those neighbors will then be determined using that neighbor.
- The vector is multiplied by a chance number between 0 and 1.
- You combine this with the existing data point to get the synthetic data point.

Figure-2 demonstrate pre and post SMOTE class balance plots of CNS cancer dataset.

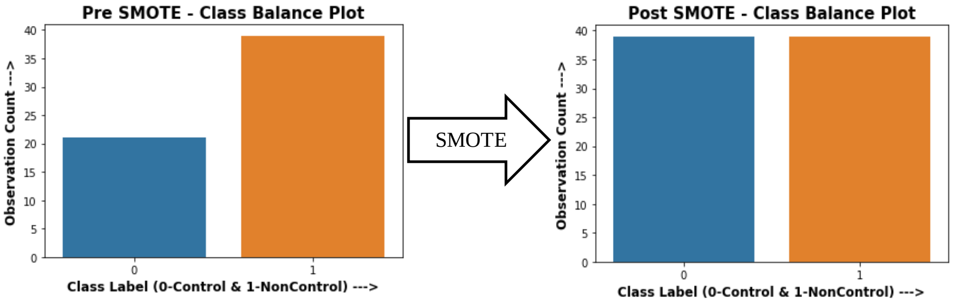


Fig.2. Class balance pre & post SMOTE

3.3 Data Sampling

Data sampling is the process of identifying and selecting data splits for model’s training and testing purpose separately. This study utilized stratified random data sampling approach withstanding frequently adopted train & test set ration of 80:20. Randomness control parameter referenced as “random_state” was preset to constant value 42 throughout experimentation process of this study. [37-38]

3.4 Feature Selection (FS)

Feature Selection phase typically deals with identifying most correlated and non-redundant features from high dimensional feature space, which typically required when dealing with high dimensional dataset i.e., image, gene or genomic & audio dataset. [39] Higher the numbers of features in the train data of ML model, higher the complexity of ML model becomes leading to overfitting issue. Thus, efficient feature selection is need of great importance during working with high-dimensional dataset. [40] Current study makes use of Maximum Relevance -Minimum Redundancy (mRMR) method of FS to select two subsets of top 1000 (mRMR-1000) & 500 (mRMR-500) features. mRMR is a ML model / classifier independent minimal-optimal FS algorithm with least time & memory consumption. A selection of features with the highest correlation to the prediction class (output/target) and the lowest correlation among themselves are often chosen using mRMR. Based on mutual information, it ranks features using the minimal-redundancy-maximal-relevance criterion. [41-42]

3.5 Model's Pre-evaluation & Base Feature Subset Selection & Model Selection

Post mRMR-based feature selection, total of 12 ML classifiers i.e., LR, SVC (Linear & RBF), NB, KNN, RF, DT, ET, GbBoost, XgBoost, AdaBoost & MLP were utilized during model training process of pre-evaluation phase of proposed methodology of current study. [30, 32] All 12 ML models were evaluated on both feature subsets (mRMR-1000 & mRMR-500) on the basis of various performance metrics i.e., accuracy, precision, recall, F1-Score & AUC-Score and feature subset exhibiting higher classification performance by all 12 ML model needs to considered for further processing as base feature subset. [36, 38] Current study identified mRMR-500 feature subset as base feature space on the basis of pre-evaluation's results.

Apart from base feature subset selection process, model selection process is also carried out concurrently during this phase of study. Top three ML model needs to identified on the basis of pre-evaluation of all 12 ML models on selected base feature subset (mRMR-500) with their basic configuration (with default parameter values). Top-3 ML models identified in current phase of this study are LR, SVC (Linear) & SVC (RBF).

3.6 Dimensionality Reduction

As a result of model's pre-evaluation & base subset selection phase, feature subset of top 500 highly correlated features (mRMR-500) was identified and selected as base feature subset for next phase of this study, which is dimensionality reduction phase. In this phase we further reduce dimensionality of feature space using Recursive Feature Elimination (RFE) method. [19-20] It is model dependent methods, which means this used to sind optimal features in accordance to model considered for prediction purpose. In RFE, as name suggest, all 500 features of mRMR-500 feature subset were investigated initially to find feature's contribution (rank) toward target feature and recursively eliminating or dropping features having low contribution (lower rank) till certain or user-defined no features left in the resulting feature space. [43-44] In current study, five feature subsets of 100, 50, 30, 10 & 5 features, were generated using RFE method, which are referenced as RFE-100, RFE-50, RFE-30, RFE-10 & RFE-5 respectively. Top-3 ML model, selected from previous phase, referred as LR, SVC (Linear) & SVC (RBF) were trained and evaluated on above mentioned five feature subsets to identify best two feature subsets for further phases of proposed methodology. Current study found RFE-100 & RFE-50 feature subsets most suitable subsets for further processing as all top three ML model demonstrated highest classification performance on these two subsets with their default parameters.

Table.1. List of selected classifiers-wise feature Subsets via RFE method

Feature Subset	Logistic Regression (LR)	Support Vector Classifier [SVC (Linear & RBF)]
RFE-5	["U37673_at", "HG3523-HT4899_s_at", "D49490_at", "X70649_at", "X93510_at"]	["HG2994-T4850_s_at", "U75276_s_at", "M12625_at", "D49490_at", "X93510_at"]
RFE-10	["U37673_at", "X64624_s_at", "HG3523-T4899_s_at", "Z31560_s_at", "D49490_at", "X70649_at", "HG384-T384_at", "M55513_s_at", "J04823_rna1_at", "X93510_at"]	["HG2994-T4850_s_at", "U75276_s_at", "M12625_at", "D49490_at", "U15008_at", "HG384-HT384_at", "S82240_at", "U29953_rna1_at", "M14328_s_at", "X93510_at"]
RFE-30	["U37673_at", "HG2994-HT4850_s_at", "X64624_s_at", "HG3523-HT4899_s_at", "U78180_at", "S71824_at", "X76302_at", "M29277_s_at", "Z31560_s_at", "M13207_at", "D49490_at", "U15008_at", "U28811_at", "HG2157-HT2227_at", "X70649_at", "HG384-HT384_at", "L33842_rna1_at", "U29953_rna1_at", "HG998-HT998_s_at", "M14328_s_at", "M55513_s_at", "X74295_at", "X02152_at", "D50663_at", "Y08409_at", "J04823_rna1_at", "U45955_at", "AB000460_at", "X93510_at", "M35878_at"]	["S76475_at", "HG2994-HT4850_s_at", "X59798_at", "U75276_s_at", "HG3523-HT4899_s_at", "X03794_s_at", "M12625_at", "X76302_at", "D43682_s_at", "M29277_s_at", "Z31560_s_at", "D49490_at", "HG4318-HT4588_s_at", "U15008_at", "X00734_at", "HG384-HT384_at", "S82240_at", "X13461_s_at", "U29953_rna1_at", "M14328_s_at", "X02152_at", "M60854_at", "L11672_r_at", "U45955_at", "M60721_at", "AB000460_at", "Z14244_at", "X93510_at", "M35878_at", "U52696_s_at"]
RFE-50	["J02611_at", "U37673_at", "HG2994-HT4850_s_at", "X64624_s_at", "X59798_at", "HG3523-HT4899_s_at", "U78180_at", "M12625_at", "S71824_at", "Y07604_at", "X76302_at", "D43682_s_at", "M29277_s_at", "Z31560_s_at", "M13207_at", "X15880_at", "D49490_at", "J00073_at", "HG4318-HT4588_s_at", "U15008_at", "U40271_s_at", "U28811_at", "HG2157-HT2227_at", "X00734_at", "X70649_at", "Z19585_at", "HG384-HT384_at", "L33842_rna1_at", "S82240_at", "U29953_rna1_at", "HG998-HT998_s_at", "D87673_at", "M14328_s_at", "X78565_at", "M55513_s_at", "X74295_at", "X02152_at", "D50663_at", "Y08409_at", "U66406_at", "J04823_rna1_at", "U45955_at", "M60721_at", "AB000460_at", "Z14244_at", "X14445_at", "X93510_at", "Z74616_s_at", "M35878_at", "U52696_s_at"]	["J02611_at", "S76475_at", "S66541_s_at", "HG2994-HT4850_s_at", "X59798_at", "U75276_s_at", "HG3523-HT4899_s_at", "X03794_s_at", "M12625_at", "Y07604_at", "X76302_at", "D43682_s_at", "M29277_s_at", "Z31560_s_at", "M25667_at", "M13207_at", "D49490_at", "HG4318-HT4588_s_at", "U15008_at", "HG2157-HT2227_at", "M21494_at", "X00734_at", "Z19585_at", "HG384-HT384_at", "S82240_at", "X13461_s_at", "U29953_rna1_at", "HG998-HT998_s_at", "M14328_s_at", "X78565_at", "U78876_at", "U43522_at", "AB001325_at", "X02152_at", "Y08409_at", "U31986_at", "M60854_at", "U66406_at", "J04823_rna1_at", "L11672_r_at", "X53331_at", "U45955_at", "M60721_at", "AB000460_at", "Z14244_at", "L16862_at", "X93510_at", "M35878_at", "HG613-HT613_at", "U52696_s_at"]

RFE-100	["L11369_at", "Z74615_at", "HG2417-HT2513_at", "J02611_at", "U37673_at", "S76475_at", "S66541_s_at", "HG2994-HT4850_s_at", "X74801_at", "X64624_s_at", "D86977_at", "X59798_at", "U75276_s_at", "L26081_at", "S78296_at", "HG3523-HT4899_s_at", "U78180_at", "X70811_at", "M13194_at", "X03794_s_at", "M12625_at", "S71824_at", "X15882_at", "Y07604_at", "U29943_s_at", "D80004_at", "X76302_at", "D43682_s_at", "M29277_s_at", "Z31560_s_at", "M25667_at", "HG4011-HT4804_s_at", "M28219_at", "U63455_at", "M13207_at", "X15880_at", "D16688_s_at", "D49490_at", "M63962_rna1_at", "J00073_at", "HG4318-HT4588_s_at", "U15008_at", "U40271_s_at", "U28811_at", "HG2157-HT2227_at", "M21494_at", "Y09305_at", "X00734_at", "X70649_at", "Z19585_at", "HG384-HT384_at", "L33842_rna1_at", "U37012_at", "S82240_at", "X13461_s_at", "U29953_rna1_at", "HG998-HT998_s_at", "D87673_at", "M14328_s_at", "D42053_at", "X78565_at", "L31881_at", "Y09836_at", "U78876_at", "X04412_at", "S69265_s_at", "M55513_s_at", "X74295_at", "L18983_at", "U43522_at", "AB001325_at", "X02152_at", "D50663_at", "Y08409_at", "U31986_at", "X54938_at", "L03840_s_at", "U66406_at", "J04823_rna1_at", "L11672_r_at", "X53331_at", "U45955_at", "U39576_at", "M60721_at", "J04760_at", "L41147_at", "L04731_at", "Z75190_s_at", "AB000460_at", "Z14244_at", "M33318_r_at", "X14445_at", "L16862_at", "X93510_at", "D12676_at", "Z74616_s_at", "M35878_at", "M55998_s_at", "HG613-HT613_at", "U52696_s_at"]	["L11369_at", "Z74615_at", "HG2417-HT2513_at", "J02611_at", "U37673_at", "S76475_at", "S66541_s_at", "HG2994-HT4850_s_at", "X74801_at", "X64624_s_at", "D86977_at", "X59798_at", "U75276_s_at", "L26081_at", "HG3523-HT4899_s_at", "U78180_at", "X03794_s_at", "M12625_at", "S71824_at", "Y07604_at", "U29943_s_at", "D80004_at", "X76302_at", "D43682_s_at", "M29277_s_at", "Z31560_s_at", "M25667_at", "HG4011-HT4804_s_at", "M28219_at", "M13207_at", "X15880_at", "D16688_s_at", "D49490_at", "M63962_rna1_at", "J00073_at", "HG4318-HT4588_s_at", "U15008_at", "U40271_s_at", "U28811_at", "HG2157-HT2227_at", "Z26876_at", "M21494_at", "D13631_s_at", "Y09305_at", "X00734_at", "L27560_at", "X70649_at", "Z19585_at", "HG384-HT384_at", "L33842_rna1_at", "S82240_at", "X13461_s_at", "U29953_rna1_at", "HG998-HT998_s_at", "D87673_at", "M14328_s_at", "X78565_at", "J03242_s_at", "S78187_at", "U78876_at", "X04412_at", "S69265_s_at", "M55513_s_at", "X74295_at", "L18983_at", "U43522_at", "AB001325_at", "X02152_at", "D50663_at", "Y08409_at", "U31986_at", "M60854_at", "X54938_at", "L03840_s_at", "U66406_at", "J04823_rna1_at", "L11672_r_at", "X53331_at", "U41813_at", "HG2663-HT2759_at", "U45955_at", "U39576_at", "U61741_at", "M60721_at", "J04760_at", "HG2810-HT2921_at", "U09770_at", "HG243-HT243_s_at", "AB000460_at", "Z14244_at", "M33318_r_at", "X14445_at", "L16862_at", "X93510_at", "D12676_at", "Z74616_s_at", "M35878_at", "M55998_s_at", "HG613-HT613_at", "U52696_s_at"]

3.7 Model Optimization

After obtaining the best ML classifier through the aforementioned stages of this study, the following stage, known as model optimization, entails determining the ideal value for each model parameter to get the best possible prediction accuracy from the used model. It is also known as an optimization procedure or hyperparameter tuning. [45] A model hyperparameter is a parameter setting that is extrinsic to the ML model and whose value is non-deterministic from training data. These parameters are widely used in techniques to help with model parameter estimates and explain important model properties including complexity and learning rate. The best hyperparameter value for a model can be found manually or by utilizing a tree-based technique, such as grid search or random search. A manual search might be time-consuming because one model may have several hyperparameters with a significant range of values. In order to find hyperparameters automatically, ML practitioners often employ grid search or random search techniques. [21, 46] The hyperparameter combination for which the ML model with considered hyperparameter values exhibit optimum classification performance is returned by both techniques, which train and assess model performance on various combinations of hyperparameter values. As it seeks parameter values over the whole parameter combination space, this study used the grid search approach of hyperparameter tuning.

Grid search method is evaluated for various hyperparameter's values of three selected ML models i.e., LR, SVC (Linear) & SVC (RBF), which were obtained during pre-evaluation phase of current study. Mapped hyperparameter search space and received optimal hyperparameters values using grid search method for above three models are enumerated in table-2.

Table.2. List of mapped hyperparameter search space and optimal hyperparameters values using Grid Search Method

Model Name	Hyperparameter Name	Hyperparameter Search Space	Optimal Hyperparameter Value
Logistic Regression (LR)	"C"	[0.1,1, 10, 100, 1000]	100
	"solver"	['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga']	newton-cg
	"penalty"	['none', 'l1', 'l2', 'elasticnet']	none
Support Vector Machine (SVC-Linear)	"C"	[0.1,1, 10, 100, 1000]	0.1
	"gamma"	[1,0.1,0.01,0.001,0.0001]	1
Support Vector Machine (SVC-RBF)	"C"	[0.1,1, 10, 100, 1000]	10
	"gamma"	[1,0.1,0.01,0.001,0.0001]	0.01

3.8 Model's Post-evaluation

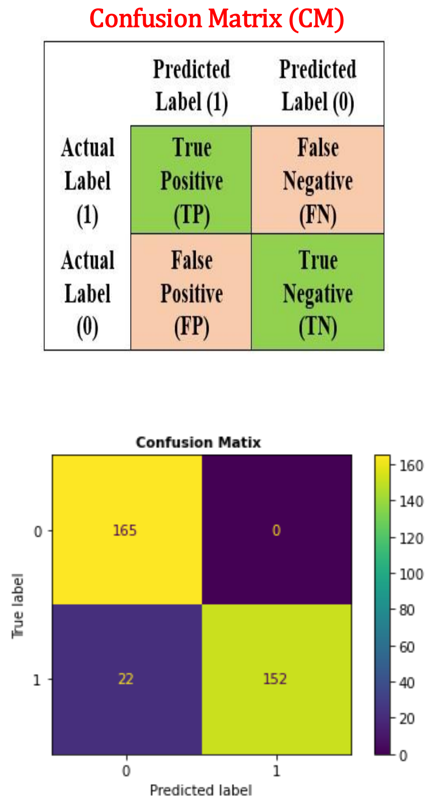
Post model optimization, all three selected ML model have been employed on both optimal feature subset (RFE-100 & RFE-50) and evaluated using different performance metrics discussed in section 3.9 to find performance increase of ML models. Although all three models already performed reasonably well on both feature subset without hyperparameter tuning and there was only a small scope for improvement, however, this study observed significant improvement in LR model's performance with tuned hyperparameter values in case of RFE-100 feature subset only.

3.9 Performance Metrics

Following the model optimization stage, ML model is put into practice, and results are generated as a class or a probability. The next stage is to use a test dataset and some appropriate performance metrics to assess the model's effectiveness during prediction of Parkinson disease. Various metrics, including accuracy, recall, F-1 score, precision, and AUC-ROC curve, were been employed in this work to evaluate the classification performance of ML models. It is crucial to pick the right metrics to assess the ML model since they have an impact on how performance is compared and monitored. [47-48] Brief introduction of these performance metrics is presented in next upcoming subsections.

3.9.1 Confusion Matrix (CM)

Confusion matrix (CM) is primarily a base performance metrics for evaluating a classification ML model. Almost every performance metric used for classification problem makes use of CM's parameters i.e., true-positive (TP), true-negative (TN), false-positive (FP) & false-negative (FN). As PD classification is a two class ("1" & "0") classification task, CM will be 2×2 matrix, where one dimension represents actual target value and second represents predicted value as demonstrated in figure-3. [49-50] The basic terminologies of CM are as follows:



Confusion Matrix (CM)

	Predicted Label (1)	Predicted Label (0)
Actual Label (1)	True Positive (TP)	False Negative (FN)
Actual Label (0)	False Positive (FP)	True Negative (TN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision(P) = \frac{TP}{TP + FP}$$

$$Recall(R) = \frac{TP}{TP + FN}$$

$$F1 - Score = \frac{2 \times P \times R}{P + R}$$

$$TPR(sensitivity) = \frac{TP}{TP + FN}$$

$$TNR(\text{specificity}) = \frac{TN}{TN + FP}$$

$$FPR = \frac{FP}{TN + FP}$$

$$FNR = \frac{FN}{TP + FN}$$

Figure.3. Confusion Matrix for binary classification problems

- **True-Positive (TP)**

  It can be described as the capacity of a model, to classify classes accurately as positive (+ve), such that if actual class is 1, the predicted class will also be 1. It is also referred as sensitivity as well as True Positive Rate (TPR) in percentage format. [47]

$$TPR(sensitivity) = \frac{TP}{TP + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

- **True Negative (TN)**

It is described as the capacity of a model, to classify classes accurately as negative (-ve), i.e., if real class is 0, then the projected class will also be 0. It also known as specificity as well as True Negative Rate (TNR) in percentage format. [48]

$$TNR(\text{specificity}) = \frac{TN}{TN + FP} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (2)$$

- **False Positive (FP)**

It represents misclassification of negative class to positive class label. In other words, the model predicts a class label of 1 while the initial class label was 0. It also known False Positive Rate (FPR) in percentage format. [49]

$$FPR = \frac{FP}{TN + FP} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (3)$$

- **False Negative (FN)**

It represents misclassification of positive class to negative class label. In other words, the model predicts a class label of 0 while the initial class label was 1. It also known False Negative Rate (FNR) in percentage format. [50]

$$FNR = \frac{FN}{TP + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (4)$$

3.9.2 Accuracy

It represents the proportion of correctly predicted examples to all the instances in the dataset. It is observed to be good performance metrics for classification only when there is a class balance in considered dataset. For imbalanced dataset it is not a recommended choice for performance evaluation of ML model. [47]

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (5)$$

3.9.3 Precision

In terms of how many of the expected positive outcomes materialized, Precision measures how accurate your model is. It is determined by dividing the number of retrieved instances by the proportion of true positive relevant instances. [49]

$$Precision(P) = \frac{TP}{TP + FP} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (6)$$

3.9.4 Recall

It is also known as sensitivity and is described as the ratio of accurately predicted positive (+ve) classes to all positive occurrences. Whenever false-negative (FN) has a significant importance, recall should be model's metric we utilize to identify our most promising model. [48]

$$Recall(R) = \frac{TP}{TP + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (7)$$

3.9.5 F1-score

Circumstances, where one is unable to assess applicability of recall & precision metrics for one's classification task, F1-score may be suitable choice for performance evaluation as it combines both recall and precision or in other context it stands for a harmonic mean of recall and precision. [49]

$$F1 - Score = \frac{2 \times P \times R}{P + R} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (8)$$

3.9.6 AUC-ROC Curve

Area under the curve (AUC) - Receiver Operating Characteristics (ROC) curve is a well-known probability curve between TPR & FPR, mostly used for classification models, where ROC is a probability curve and AUC is a measure of separability. This curve demonstrates how effective the classification model is at differentiating between classes. The ROC curve's y-axis displays the true positive rate, while the x-axis displays the false positive rate. AUC ranges in value from 0 to 1. The model performs extraordinarily well in terms of classification when the AUC is near to 1, but performs poorly in terms of separability and is unable to separate data when the AUC is close to 0.5. [47-51]

4. Results & Discussion

Experimentation process involved data exploration initially, during which class imbalance problem has been identified and resolved via SMOTE oversampling method. [51] This study pre-evaluated 12 ML models, including LR, KNN, SVC (Linear & RBF), NB, DT, RF, ET, GbBoost, XgBoost, AdaBoost & MLP, using both datasets D1 (Unbalanced CNS dataset) and D2 (Balanced CNS dataset), in order to see how the performance of the used ML models improved after oversampling. [47, 52-54] All models were used at this point with their default settings alone. Results of pre-evaluation performance of different ML classifiers are as presented in table-3.

Table.3. Pre-evaluation performance of different ML models

Dataset	**D1**	**D2**	**D1**	**D2**	**D1**	**D2**	**D1**	**D2**	**D1**	**D2**
ML Model Name	**Accuracy**		**Precision**		**Recall**		**F1-Score**		**AUC**	
LR	0.611	0.875	0.63	0.9	0.61	0.88	0.56	0.87	0.6625	0.9792
KNN	0.611	0.625	0.68	0.79	0.61	0.62	0.59	0.56	0.575	0.5799
SVC (Linear)	0.556	0.917	0.54	0.93	0.56	0.92	0.52	0.92	0.525	0.9167
SVC (RBF)	0.556	0.833	0.31	0.88	0.56	0.83	0.4	0.83	0.5	0.8333
NB	0.667	0.708	0.67	0.75	0.67	0.71	0.67	0.7	0.7	0.7917
DT	0.5	0.708	0.48	0.71	0.5	0.71	0.47	0.71	0.5375	0.75
RF	0.5	0.708	0.49	0.71	0.5	0.71	0.5	0.71	0.7188	0.875
ET	0.556	0.917	0.57	0.93	0.56	0.92	0.56	0.92	0.5625	0.9167
GbBoost	0.611	0.75	0.61	0.75	0.61	0.75	0.61	0.75	0.6062	0.7778
XGBoost	0.611	0.625	0.61	0.63	0.61	0.62	0.61	0.62	0.5625	0.7639
AdaBoost	0.556	0.667	0.54	0.67	0.56	0.67	0.52	0.66	0.4	0.8472
MLP	0.611	0.75	0.64	0.78	0.61	0.75	0.61	0.74	0.5812	0.875

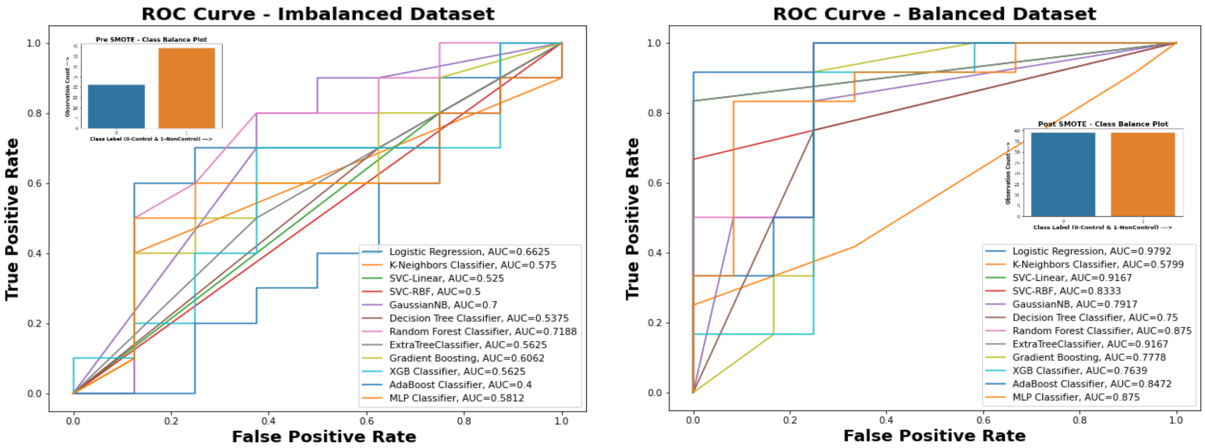


Figure.4. Model's performance on CNS caner dataset (pre & post oversampling)

Figure 4 shows the AUC-ROC (AUROC) curve for the prediction performance of 12 ML models on balanced (D2) and unbalanced (D1) datasets. Greater the area under the curve, the lower the ML model's rate of misclassification. As seen above, practically all ML models have shown improved performance in terms of several performance-metrics, indicating that ML models work effectively when the given dataset has predictor class balance. By lowering the number of misclassifications and increasing the rate of accurate classification by ML model, this assures impartial performance of applied ML model. [55]

Concurrent to pre-evaluation phase itself, feature selection (FS) phase has been implemented to identify three feature subsets including best 500 & 1000 features using mRMR FS method. Both mRMR-500 & mRMR-1000 feature subsets have been investigated during this study to identify most favorable feature subset out of these two on the basis of classification accuracy of 12 applied ML model (default parameter configuration). Investigation results presented in table-4 and figure-5 depicts suitability of feature subset mRMR-500 as well as identification of top-3 best performer ML models (LR, SVC (Linear) & SVC (RBF)) for further experimentation of current study.

Table.4. Performance of ML model (default parameters) mRMR-500 & mRMR-1000 feature subsets

Classifier Name	Accuracy			Precision			Recall			F1-Score			AUC		
	ALL	mRMR-500	mRMR-1000	ALL	mRMR-500	mRMR-1000	ALL	mRMR-500	mRMR-1000	ALL	mRMR-500	mRMR-1000	ALL	mRMR-500	mRMR-1000
LR	0.875	**0.958**	0.9167	0.9	**0.96**	0.93	0.88	**0.96**	0.92	0.87	**0.96**	0.92	0.9792	**1**	1
KNN	0.625	0.792	0.708	0.79	0.85	0.82	0.62	0.79	0.71	0.56	0.78	0.68	0.5799	0.9688	0.875
SVC (Linear)	0.917	**0.958**	0.958	0.93	**0.96**	0.96	0.92	**0.96**	0.96	0.92	**0.96**	0.96	0.9167	**0.9583**	0.9583
SVC (RBF)	0.833	**0.958**	0.875	0.88	**0.96**	0.9	0.83	**0.96**	0.88	0.83	**0.96**	0.87	0.8333	**0.9583**	0.875
NB	0.708	0.875	0.9167	0.75	0.88	0.93	0.71	0.88	0.92	0.7	0.87	0.92	0.7917	0.9514	0.9375
DT	0.708	0.708	0.75	0.71	0.72	0.75	0.71	0.71	0.75	0.71	0.7	0.75	0.75	0.8333	0.75
RF	0.708	0.708	0.75	0.71	0.75	0.75	0.71	0.71	0.75	0.71	0.7	0.75	0.875	0.9792	0.9653
ET	0.917	0.792	0.667	0.93	0.79	0.69	0.92	0.79	0.67	0.92	0.79	0.66	0.9167	0.8333	0.7917
GbBoost	0.75	0.875	0.875	0.75	0.88	0.88	0.75	0.88	0.88	0.75	0.87	0.87	0.7778	0.9583	0.8229
XGBoost	0.625	0.792	0.833	0.63	0.81	0.84	0.62	0.79	0.83	0.62	0.79	0.83	0.7639	0.9792	0.8889
AdaBoost	0.667	0.792	0.792	0.67	0.81	0.81	0.67	0.79	0.79	0.66	0.79	0.79	0.8472	0.9583	0.9722
MLP	0.75	0.9167	0.9167	0.78	0.93	0.93	0.75	0.92	0.92	0.74	0.92	0.92	0.875	1	1

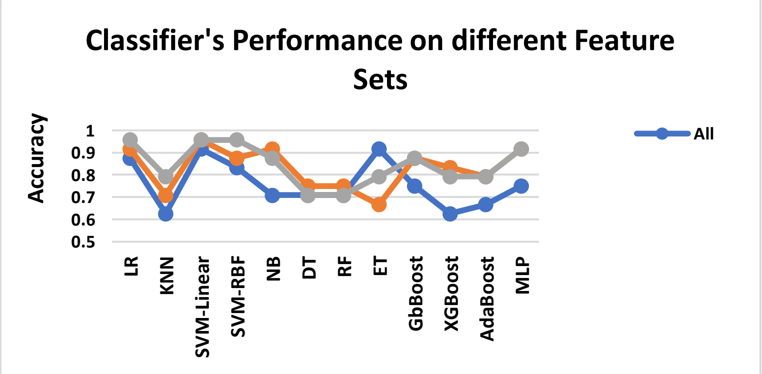


Figure 5. Performance of various ML models on different mRMR feature sets

Post selection of most favorable feature subset (mRMR-500) & ML classifiers (LR & SVC (Linear & RBF), next phase of current study was to further reduce dimensionality feature subset mRMR-500 using model dependent feature selection method called recursive feature elimination (RFE) to obtain best RFE feature subset on the basis of identified three top ML model's evaluation. [56-57] Total of five feature subsets referred as RFE-100, RFE-50, RFE-30, RFE-10 & RFE-5, consisting of features count of 100, 50, 30, 10 & 5 respectively, has been obtained for ML model's evaluations in this phase of study (refer table-1 for details). Table-5 demonstrate performance of default parametric LR & SVC (Linear & RBF) models on different feature subsets generated classifier wise by RFE algorithm.

Table.5. Performance of LR, SVC (Linear/RBF) models (default parameters) on RFE feature subsets

Classifier Name	ALL Feature					RFE-100					RFE-50				
	Accuracy	Precision	Recall	F1-Score	AUC	Accuracy	Precision	Recall	F1-Score	AUC	Accuracy	Precision	Recall	F1-Score	AUC
LR	0.875	0.9	0.88	0.87	0.9792	**0.9583**	**0.96**	**0.96**	**0.96**	**1**	**0.958**	**0.96**	**0.96**	**0.96**	**1**
SVC (Linear)	**0.917**	**0.93**	**0.92**	**0.92**	**0.9167**	**0.958**	**0.96**	**0.96**	**0.96**	**0.9583**	**0.958**	**0.96**	**0.96**	**0.96**	**0.9583**

SVC (RBF)	0.833	0.88	0.83	0.83	0.8333	0.8333	0.84	0.83	0.83	0.8333	**0.958**	**0.96**	**0.96**	**0.96**	**0.9583**

Classifier Name	**RFE-30**					**RFE-10**					**RFE-5**				
	Accuracy	**Precision**	**Recall**	**F1-Score**	**AUC**	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**AUC**	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**AUC**
LR	**0.993**	**0.99**	**0.98**	**0.99**	**1**	**0.958**	**0.96**	**0.96**	**0.96**	**0.9583**	0.667	0.69	0.67	0.66	0.7847
SVC (Linear)	0.917	0.93	0.92	0.92	0.9167	0.75	0.76	0.75	0.75	0.75	0.708	0.72	0.71	0.7	0.708
SVC (RBF)	0.875	0.88	0.88	0.87	0.875	0.833	0.83	0.83	0.82	0.8333	**0.75**	**0.83**	**0.75**	**0.73**	**0.75**

As presented and observed in table-5, on the basis of performance evaluation of LR & SVC (Linear & RBF) models, two subsets RFE-100 & RFE-50 has shown significantly better performance in all three applied models. Thus both of these feature subsets were locked for further phases of this study. Till this phase all ML models utilized in this study were employed and evaluated using model's default parameters values. In next phase, model optimization, of this study, all three models were optimized & tuned using best hyperparameter values obtained from GridSearchCV () method with 10-cross-validation (CV). [58-59] Different search space values from various models were sent to the grid search algorithm, and after careful processing, the best feature values were discovered. List of models-wise hyperparameter names, supplied search space values and received optimal hyperparameter values by grid search method is presented in table-2.

After obtaining tuned hyperparameters values, all three models were employed on RFE-100 & RFE-50 feature subsets to asses possible performance improvement in classification performance of these three ML models and compared using accuracy, recall, F1-score, precision & AUC-score. Evaluation results of this post evaluation of these models have been presented in table-6 and prediction performance of same models in terms of AUC-ROC (AUROC) curve is shown in figure-6.

Table-6. Performance of LR, SVC (Linear/RBF) models (tuned parameter) models on feature subsets

Classifier Name	**Feature Subset**	**Pre-Hyperparameter Tuning**					**Post-Hyperparameter Tuning**				
		Accuracy	**Precision**	**Recall**	**F1-Score**	**AUC**	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**AUC**
LR	**RFE-50**	0.958	0.96	0.96	0.96	1	0.958	0.96	0.96	0.96	1
SVC (Linear)		0.958	0.96	0.96	0.96	0.9583	0.958	0.96	0.96	0.96	0.9583
SVC (RBF)		0.916	0.95	0.94	0.95	0.9583	**0.958**	**0.96**	**0.96**	**0.96**	**0.9583**
LR	**RFE-100**	0.9583	0.96	0.96	0.96	1	**0.996**	**0.99**	**0.98**	**0.99**	**1**
SVC (Linear)		0.958	0.96	0.96	0.96	1	**0.987**	**0.98**	**0.97**	**0.98**	**1**
SVC (RBF)		0.8333	0.84	0.83	0.83	0.8333	**0.958**	**0.96**	**0.96**	**0.96**	**0.9583**

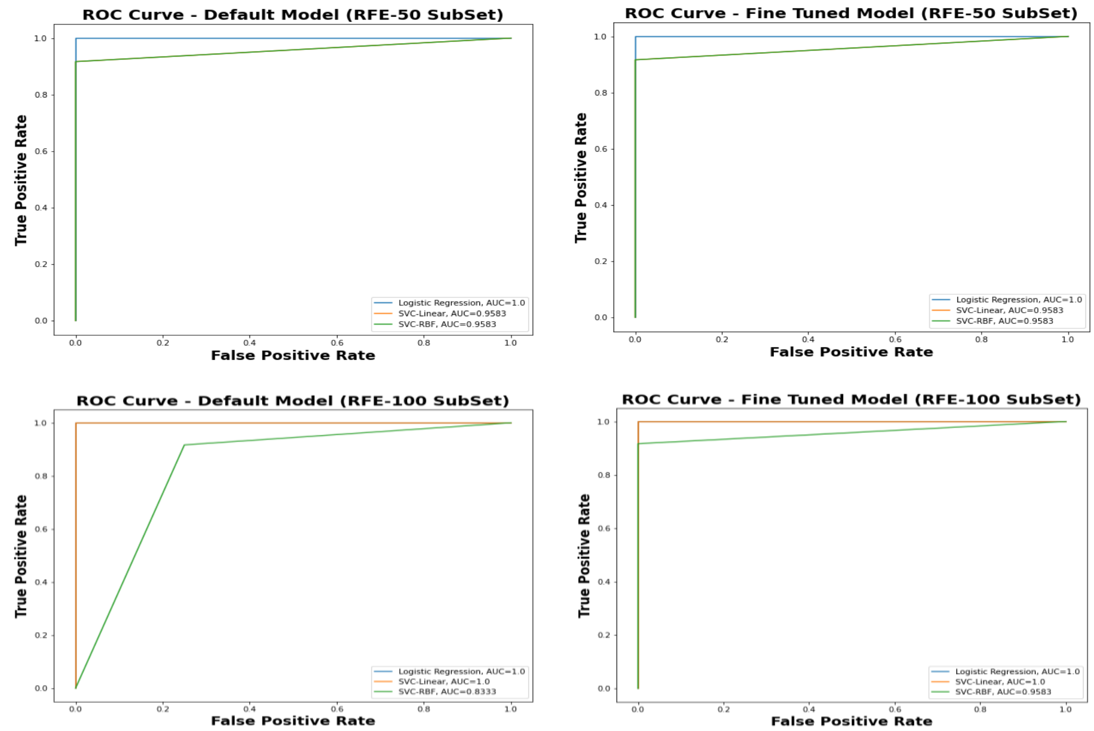


Figure 6. Performance evaluation of LR, SVC (Linear & RBF) models on RFE-100 & RFE-50 feature sets

On further evaluation, proposed method, referred as Logistic regression (LR) with minimum redundancy maximum relevance (mRMR)recursive feature elimination (RFE) - [LR + mRMR/RFE], is compared with past reported work on same CNS cancer gene expression dataset is presented in table-7, it is observed that diagnosis performance of proposed model found to be significantly better than performance of employed models, reported in recent studies.

Table.7. Comparative Analysis of various models for CNS cancer detection using gene expression data.

ML Model	Proposed by	Accuracy (%)
SVM (Linear) + T-test	Gunavathi. C. et.al. [22]	81.25
SGA + IG	Salem, H. et.al. [23]	86.67
MLP + CFS	Arslan. M.T. et. al. [04]	97.60
SVM (RBF) + SDAE	Danaee, P. et.al. [24]	96.26
SVM (RBF) + PCA	Adiwijaya. Et.al.[25]	93.30
MLP + SA	Koul. N. et.al. [26]	96
RF + IG/GA	Al-Obeidat. et. al. [27]	97.30
SVM + PCA	Kabir MF. et.al. [28]	97.80
LR + mRMR/RFE	**Current Study**	**99.6**

As presented in table-6, LR + mRMR/RFE model found to be reasonably best model among other ML-based model employed in recent past by previous researchers, on same CNS cancer gene expression dataset used in current study. Observations from tables & figures shows dominance of Logistic Regression (LR) and Support Vector Classifier (SVC-Linear) over all rest applied ML models throughout current study's experimentation process. Observation from table-6 show a significant improvement in model's performance in case on RFE-100 feature subset, however, in case of RFE-50 feature subset no improvement has been witnessed after hyperparameter optimization except minor enhancement in SVC (RBF) model. Which represents better & more candidature of favorable feature subset to be used for CNS cancer detection task using current gene expression dataset. In current study LR-based ML model demonstrated highest classification accuracy up to 99.6%, precision of 0.99, recall of 0.98, F1-score of 0.99 and AUC-Score of 1.0 on RFE-100 feature subset with optimized hyperparameter values obtained through Grid Search method. On further analysis, this study demonstrated that in case RFE-30, RFE-10 & RFE-5 feature subsets performance of applied ML models had been degraded to certain even with default parameter values of models, signifying lower contribution to efficient diagnosis decision of CNS cancer due to higher information loss with lower no of features. Also, this study shows cased RFE-100 feature subset as optimal feature subset to work upon for CNS cancer classification using current gene expression dataset.

5. Conclusion

Gene Expression based early diagnosis of CNS cancer is of utmost significance now a days in saving CNS cancer patients life beforehand as delayed diagnosis may be critical to avoid patient's mortality. Gene expression analysis applications for developing predictive diagnosis and monitoring models have gained more attention in recent past and significant amount of progress has been witnessed in gene microarray analysis methodologies recently by researchers. This research aims to evaluate the effectiveness of several ML-based classification methods in CNS cancer diagnosis task using gene expression microarray data. A gene microarray dataset was used to apply 12 ML-based classifiers, and several assessment criteria were compared using visualization and statistical analysis. In current study, the issue of CNS cancer diagnosis is handled using ML techniques, and multiple ML models have been utilized for its detection. By analyzing the gene expression data, this study's primary goal is to demonstrate the CNS cancer's diagnosis using various ML models using efficient feature selection (FS) methods (RFE & mRMR) and model optimization by grid search method. Experimentation finding suggested that Logistic Regression (LR) & Support Vector Classifier (SVC) models performs better in than any other applied classifiers during current study. Moreover, LR model outperformed all other applied models with classification accuracy of 99.6%, precision of 0.99, recall of 0.98, F1-score of 0.99 and AUC-Score of 1.0 on RFE-100 feature subset. Sample size was been limitation of this study, however, even with this limited sample size LR-model demonstrated outstanding classification performance during CNS cancer diagnosis task. Sample size enhancement will remains future scope of this study, which is subject to data availability issue due to privacy concern of CNS cancer patients.

Funding

This work is not funded by any research agency or organization during this study.

Acknowledgements

Author is thankful to all my senior and colleagues who helped me during this article writing and also to authors, whose article we have utilized in this study.

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1]. Centers for Disease Control and Prevention. (2022). An Update on Cancer Deaths in the United States. Retrieved 2023, from https://www.cdc.gov/cancer/dcpc/research/update-on-cancerdeaths/index.htm.

[2]. Alrefai, N., Ibrahim, O., Shehzad, H. M. F., Altigani, A., Abu-ulbeh, W., Alzaqebah, M., & Alsmadi, M. K. (2022). An integrated framework based deep learning for cancer classification using microarray datasets. Journal of Ambient Intelligence and Humanized Computing, 1-12.

[3]. American Cancer Society. (2022). Cancer Facts and Figs. 2022. Retrieved 2023, fromhttps://www.cancer.org/research/cancer-facts-statistics/all-cancer-facts-figures/cancer-facts-figures2022.html.

[4]. Arslan, M. T., & Kalinli, A. (2016). A comparative study of statistical and artificial intelligence-based classification algorithms on central nervous system cancer microarray gene expression data. International Journal of Intelligent Systems and Applications in Engineering, 4(Special Issue-1), 78-81.

[5]. International Agency for Research on Cancer. (2020). Cancer Today. Retrieved 2022, from https://gco.iarc.fr/today/data/factsheets/cancers/31-Brain-central-nervous-system-fact-sheet.pdf.

[6]. Nath, G., Coursey, A., Ekong, J., Rastegari, E., Sengupta, S., Dag, A. Z., & Delen, D. (2022). Determining the Temporal Factors of Survival Associated with Brain and Nervous System Cancer Patients: A Hybrid Machine Learning Methodology.

[7]. Farmanfarma, K. K., Mohammadian, M., Shahabinia, Z., Hassanipour, S., & Salehiniya, H. (2019). Brain cancer in the world: an epidemiological review. World Cancer Research Journal, 6(5).

[8]. National Cancer Institute. (2021). Cancer Stat Facts: Brain and Other Nervous System Cancer. Retrieved 2022, from Surveillance, Epidemiology, and End Results (SEER) Program: https://seer.cancer.gov/statfacts/html/brain.html.

[9]. Dasgupta, A., Gupta, T., & Jalali, R. (2016). Indian data on central nervous tumors: A summary of published work. South Asian journal of cancer, 5(03), 147-153.

[10]. Liu, C., & Zong, H. (2012). Developmental origins of brain tumors. Current opinion in neurobiology, 22(5), 844-849.

[11]. Gour, G. B., Syed, A. H., & Gudur, B. K. (2022, October). Automated Brain Cancer Detection by Ensemble Learning Approach. In 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon) (pp. 1-6). IEEE.

[12]. Australian Institute of Health and Welfare. (2017). Brain and other central nervous system cancers. Canberra.

[13]. Akhavan, M., & Hasheminejad, S. M. H. (2023). A two-phase gene selection method using anomaly detection and genetic algorithm for microarray data. Knowledge-Based Systems, 110249.

[14]. Hue, Susan Swee-Shan, et al. "Tissue-Specific microRNA Expression Profiling to Derive Novel Biomarkers for the Diagnosis and Subtyping of Small B-Cell Lymphomas." Cancers 15.2 (2023): 453.

[15]. Zolfaghari, B., Mirsadeghi, L., Bibak, K., & Kavousi, K. (2023). Cancer Prognosis and Diagnosis Methods Based on Ensemble Learning. ACM Computing Surveys.

[16]. Mahoto, N. A., Shaikh, A., Sulaiman, A., Al Reshan, M. S., Rajab, A., & Rajab, K. (2023). A machine learning based data modeling for medical diagnosis. Biomedical Signal Processing and Control, 81, 104481.

[17]. Bhatt, M., & Shende, P. (2023). Advancement in Machine Learning: A Strategic Lookout from Cancer Identification to Treatment. Archives of Computational Methods in Engineering, 1-16.

[18]. Maurya, S., Tiwari, S., Mothukuri, M. C., Tangeda, C. M., Nandigam, R. N. S., & Addagiri, D. C. (2023). A review on recent developments in cancer detection using Machine Learning and Deep Learning models. Biomedical Signal Processing and Control, 80, 104398.

[19]. Qu, K., Xu, J., Han, Z., & Xu, S. (2023). Maximum relevance minimum redundancy-based feature selection using rough mutual information in adaptive neighborhood rough sets. Applied Intelligence, 1-20.

[20]. Ding, X., Yang, F., & Ma, F. (2022). An efficient model selection for linear discriminant function-based recursive feature elimination. Journal of Biomedical Informatics, 129, 104070.

[21]. Guido, R., Groccia, M. C., & Conforti, D. (2022, March). Hyper-Parameter Optimization in Support Vector Machine on Unbalanced Datasets Using Genetic Algorithms. In Optimization in Artificial Intelligence and Data Sciences: ODS, First Hybrid Conference, Rome, Italy, September 14-17, 2021 (pp. 37-47).

[22]. Gunavathi, C., & Premalatha, K. (2014). Performance analysis of genetic algorithm with kNN and SVM for feature selection in tumor classification. International Journal of Computer and Information Engineering, 8(8), 1490-1497.

[23]. Salem, H., Attiya, G., & El-Fishawy, N. (2017). Classification of human cancer diseases by gene expression profiles. Applied Soft Computing, 50, 124-134.

[24]. Danaee, P., Ghaeini, R., & Hendrix, D. A. (2017). A deep learning approach for cancer detection and relevant gene identification. In Pacific symposium on biocomputing 2017 (pp. 219-229).

[25]. Adiwijaya, W. U., Lisnawati, E., Aditsania, A., & Kusumo, D. S. (2018). Dimensionality reduction using principal component analysis for cancer detection based on microarray data classification. Journal of Computer Science, 14(11), 1521-1530.

[26]. Koul, N., & Manvi, S. S. (2022). Feature Selection from Gene Expression Data Using Simulated Annealing and Partial Least Squares Regression Coefficients. Global Transitions Proceedings.

[27]. Al-Obeidat, F., Tubaishat, A., Shah, B., & Halim, Z. (2020). Gene encoder: a feature selection technique through unsupervised deep learning-based clustering for large gene expression data. Neural Computing and Applications, 1-23.

[28]. Kabir, M. F., Chen, T., & Ludwig, S. A. (2023). A performance analysis of dimensionality reduction algorithms in machine learning models for cancer prediction. Healthcare Analytics, 3, 100125.

[29]. Pomeroy, S. L., Tamayo, P., Gaasenbeek, M., Sturla, L. M., Angelo, M., McLaughlin, M. E., ... & Golub, T. R. (2002). Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature, 415(6870), 436-442.

[30]. Painuli, D., & Bhardwaj, S. (2022). Recent advancement in cancer diagnosis using machine learning and deep learning techniques: A comprehensive review. Computers in Biology and Medicine, 105580.

[31]. Shabbir, S., Asif, M. S., Alam, T. M., & Ramzan, Z. (2021). Early prediction of malignant mesothelioma: an approach towards non-invasive method. Current Bioinformatics, 16(10), 1257-1277.

[32]. Painuli, D., Mishra, D., Bhardwaj, S., & Aggarwal, M. (2021). Forecast and prediction of COVID-19 using machine learning. In Data Science for COVID-19 (pp. 381-397). Academic Press.

[33]. Chatterjee, S., Mastalerz, M., Drobniak, A., & Karacan, C. Ö. (2022). Machine learning and data augmentation approach for identification of rare earth element potential in Indiana Coals, USA. International Journal of Coal Geology, 259, 104054.

[34]. Mishra, D., & Painuli, D. (2016). Rule Based Expert System for Medical Diagnosis-A Review. International Journal of Engineering Technology, Management, and Applied Sciences (IJETMAS), 4(12), 167-172.

[35]. Mohammedqasim, H., Mohammedqasem, R. A., Ata, O., & Alyasin, E. I. (2022). Diagnosing Coronary Artery Disease on the Basis of Hard Ensemble Voting Optimization. Medicina, 58(12), 1745.

[36]. Mishra, D., Nirvikar., Ahuja N. and Painuli, D. (2018). Fuzzy Expert System to diagnose Psoriasis Disease. International Journal of Computer Science and Information Security (IJCSIS), 16(9).

[37]. Li, C. Y., Chen, Z., Tse, T. K., Weerasuriya, A. U., Zhang, X., Fu, Y., & Lin, X. (2022). A parametric and feasibility study for data sampling of the dynamic mode decomposition: range, resolution, and universal convergence states. Nonlinear Dynamics, 107(4), 3683-3707.

[38]. Painuli, D., Mishra, D., Bhardwaj, S., & Aggarwal, M. (2020). Fuzzy rule based system to predict COVID19-a deadly virus. way, 3(4), 5.

[39]. Rostami, M., Forouzandeh, S., Berahmand, K., Soltani, M., Shahsavari, M., & Oussalah, M. (2022). Gene selection for microarray data classification via multi-objective graph theoretic-based method. Artificial Intelligence in Medicine, 123, 102228.

[40]. Wang, P., Xue, B., Liang, J., & Zhang, M. (2022). Differential evolution-based feature selection: A niching-based multi-objective approach. IEEE Transactions on Evolutionary Computation.

[41]. Wang, T., Wang, H., Deng, J., Zhang, D., Feng, J., & Chen, B. (2023). Feature generation and multi-sequence fusion based deep convolutional network for breast tumor diagnosis with missing MR sequences. Biomedical Signal Processing and Control, 82, 104536.

[42]. Balakrishnan, K., & Dhanalakshmi, R. (2022). Feature selection in high- dimensional microarray cancer datasets using an improved equilibrium optimization approach. Concurrency and Computation: Practice and Experience, 34(28), e7381.

[43]. McCague, C., Ramlee, S., Reinius, M., Selby, I., Hulse, D., Piyatissa, P., ... & Woitek, R. (2023). Introduction to radiomics for a clinical audience. Clinical Radiology, 78(2), 83-98.

[44]. Sucharita, S., Sahu, B., & Swarnkar, T. (2022). Comparative Analysis of State-Of-the-Art Classifier with CNN for Cancer Microarray Data Classification. In Intelligent and Cloud Computing: Proceedings of ICICC 2021 (pp. 533-543). Singapore: Springer Nature Singapore.

[45]. Belete, D. M., & Huchaiah, M. D. (2022). Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results. International Journal of Computers and Applications, 44(9), 875-886.

[46]. Houssein, E. H., Hassan, H. N., Al-Sayed, M. M., & Nabil, E. (2022). Intelligent Computational Models for Cancer Diagnosis: A Comprehensive Review. Integrating Meta-Heuristics and Machine Learning for Real-World Optimization Problems, 25-50.

[47]. Painuli, D. et al. "Machine Learning based Model to combat Covid19.", International Journal of Information Technology and Electrical Engineering 09.04 (2020): 33-40.

[48]. Cho, Won Ki, et al. "Diagnostic accuracies of laryngeal diseases using a convolutional neural network‐ based image classification system." The Laryngoscope 131.11 (2021): 2558-2566.

[49]. Ali, K., Shaikh, Z. A., Khan, A. A., & Laghari, A. A. (2022). Multiclass skin cancer classification using EfficientNets–a first step towards preventing skin cancer. Neuroscience Informatics, 2(4), 100034.

[50]. Oladimeji, O. O., & Oladimeji, O. (2020). Predicting survival of heart failure patients using classification algorithms. JITCE (Journal of Information Technology and Computer Engineering), 4(02), 90-94.

[51]. Tanha, J., Abdi, Y., Samadi, N., Razzaghi, N., & Asadpour, M. (2020). Boosting methods for multi-class imbalanced data classification: an experimental review. Journal of Big Data, 7(1), 1-47.

[52]. Iqbal, S., Imran, A., & Adnan, M. (2022). Breast Tumor Detection using Machine Learning Boosting Classifiers. Journal of Computing & Biomedical Informatics, 4(01), 118-131.

[53]. Chabalala, Y., Adam, E., & Ali, K. A. (2023). Exploring the Effect of Balanced and Imbalanced Multi-Class Distribution Data and Sampling Techniques on Fruit-Tree Crop Classification Using Different Machine Learning Classifiers. Geomatics, 3(1), 70-92.

[54]. Yang, B., Bao, W., Chen, B., & Song, D. (2022). Single_cell_GRN: gene regulatory network identification based on supervised learning method and Single-cell RNA-seq data. BioData Mining, 15(1), 1-18.

[55]. Jahan, S., Islam, M. S., Islam, L., Rashme, T. Y., Prova, A. A., Paul, B. K., ... & Mosharof, M. K. (2021). Automated invasive cervical cancer disease detection at early stage through suitable machine learning model. SN Applied Sciences, 3, 1-17.

[56]. Priyanka, A., & Ganesan, K. (2022). Severity estimation of brainstem in dementia MR images using moth flame optimized segmentation and fused deep feature selection. Neural Computing and Applications, 1-12.

[57]. Mostafiz, R., Uddin, M. S., Reza, M. M., & Rahman, M. M. (2022). Covid-19 detection in chest X-ray through random forest classifier using a hybridization of deep CNN and DWT optimized features. Journal of King Saud University-Computer and Information Sciences, 34(6), 3226-3235.

[58]. Taslim, T., Sabna, E., & Ningsih, K. W. (2022). Optimization of K Value at K Nearest Neighbor for Classification and Prediction of Healing in Covid-19 Patient. Journal of Pharmaceutical Negative Results, 13(4), 1699-1708.

[59]. Panda, C., Mishra, A. K., Dash, A. K., & Nawab, H. (2022). Predicting and explaining severity of road accident using artificial intelligence techniques, SHAP and feature analysis. International Journal of Crashworthiness, 1-16.